



下载APP



01 | PyTorch：网红中的顶流明星

2021-10-11 方远

《PyTorch深度学习实战》

课程介绍 >



讲述：方远

时长 13:37 大小 12.48M



你好，我是方远。

从这节课开始，我们正式进入 PyTorch 基础篇的学习。

在基础篇中，我们带你了解 PyTorch 的发展趋势与框架安装方法，然后重点为你讲解 NumPy 和 Tensor 的常用知识点。

掌握这些基础知识与技巧，能够让你使用 PyTorch 框架的时候更高效，也是从头开始学习机器学习与深度学习迈出的第一步。磨刀不误砍柴工，所以通过这个模块，我们的目标做好学习的准备工作。



今天这节课，我们先从 PyTorch 的安装和常用编程工具说起，先让你对 PyTorch 用到的语言、工具、技术做到心里有数，以便更好地开启后面的学习之旅。

PyTorch 登场

为什么选择 PyTorch 框架，我在开篇词就已经说过了。从 19 年起，无论是学术界还是工程界 PyTorch 已经霸占了半壁江山，可以说 PyTorch 已经是现阶段的主流框架了。

这里的 Py 我们不陌生，它就是 Python，那 Torch 是什么？从字面翻译过来是一个“火炬”。



什么是火炬呢？其实这跟 TensorFlow 中的 Tensor 是一个意思，我们可以把它看成是**能在 GPU 中计算的矩阵**。

那 PyTorch 框架具体是怎么用的呢？说白了就是一个计算的工具。借助它，我们就能用计算机完成复杂的计算流程。

但是我们都知，机器跟人类的“语言”并不相通，想要让机器替我们完成对数据的复杂计算，就得先把数据翻译成机器能够理解的内容。无论是图像数据、文本数据还是数值数据，都要转换成矩阵才能进行后续的变化和运算。

搞定了读入数据这一步，我们就要靠 PyTorch 搞定后面各种复杂的计算功能。这些所有的计算功能，包括了从前向传播到反向传播，甚至还会涉及其它非常复杂的计算，而这些计算统统要交给 PyTorch 框架实现。

PyTorch 会把我们需要计算的矩阵传入到 GPU（或 CPU）当中，在 GPU（或 CPU）中实现各种我们所需的计算功能。**因为 GPU 做矩阵运算比较快，所以在神经网络中的计算一般都首选使用 GPU，但对于学习来说，我们用 CPU 就可以了。**

而我们要做的就是，设计好整个任务的流程、整个网络架构，这样 PyTorch 才能顺畅地完成后面的计算流程，从而帮我们正确地计算。

安装 PyTorch 及其使用环境

在 PyTorch 安装之前，还有安装 Python3 以及 pip 这些最基础的操作，这些你在网上随便搜一下就能找到，相信你可以独立完成。

这里我直接从安装 PyTorch 开始说，PyTorch 安装起来非常非常简单，方法也有很多，这里我们先看看最简单的方法：使用 pip 安装。

使用 pip 安装 PyTorch

CPU 版本安装：

[复制代码](#)

```
1 # Linux
2 pip install torch==1.9.0+cpu torchvision==0.10.0+cpu torchaudio==0.9.0 -f http
3 # Mac & Windows
4 pip install torch torchvision torchaudio
```

GPU 版本安装：（默认 CUDA 11.1 版本）

[复制代码](#)

```
1 # Linux & Windows
2 pip install torch==1.9.0+cu111 torchvision==0.10.0+cu111 torchaudio==0.9.0 -f
```

我们只需要将上面的命令复制到计算机的命令行当中，即可实现快速安装。

这里有两个版本，一个 GPU 版本，一个 CPU 版本。建议你最好选择安装 GPU 版本，也就是说我们的硬件设备中最好有英伟达独立显卡。用 GPU 训练深度学习模型是非常快速的，所以在实际项目中都是使用 GPU 来训练模型。

但是如果说大家手里没有供开发使用的英伟达 GPU 显卡的话，那么安装 CPU 版本也是可以的，在学习过程中，CPU 也足够让我们的小实验运行起来。

另外，安装 GPU 版本前，需要安装对应版本的 CUDA 工具包。我们可以到 [英伟达官网](#)，选择相应操作系统的 CUDA 工具包，进行下载与安装。硬件设备中无英伟达显卡的，可以略过这部分。

目前 PyTorch 的稳定版本是 1.9.0，后续如果 PyTorch 的版本升级更新了，我们再将命令中的版本号稍作修改就可以了。

其它方法安装 PyTorch

[这里](#)是 PyTorch 的官网，在页面如下图所示的位置，我们可以看到有一些配置选项和安装命令。

PyTorch Build	Stable (1.9.0)		Preview (Nightly)		LTS (1.8.2)	
Your OS	Linux		Mac		Windows	
Package	Conda	Pip	LibTorch		Source	
Language	Python			C++ / Java		
Compute Platform	CUDA 10.2	CUDA 11.1	ROCm 4.2 (beta)		CPU	
Run this Command:	# Follow instructions at this URL: https://github.com/pytorch/pytorch#from-source					

我们可以根据页面上的指引，依次选择 PyTorch 的版本、你的操作系统、安装方式、编程语言以及计算平台，然后根据最下方的执行命令进行安装即可。

值得注意的是，**Mac 的操作系统只能安装 CPU 版本**。我尝试下来最简单的方式，还是使用 pip 来安装。

验证是否安装成功

你在终端中输入“python”，就可以进入到 Python 交互模式。

首先输入如下代码，如果没有报错，就意味着 PyTorch 已经顺利安装了。

```
1 import torch
```

[复制代码](#)

接下来，输入下面的代码，如果输出结果是 “True”，意味着你可以使用 GPU。这行代码的意思是检测 GPU 是否可用。

```
1 torch.cuda.is_available()
```

[复制代码](#)

这里你也许会有疑问，为什么我安装的明明是 GPU 版本，但是代码却返回了 “False”，显示 GPU 不可用呢？

对于这个问题，我们依次按照下面的步骤进行检查。

1. 检查计算机上是否有支持 CUDA 的 GPU。

首先查看电脑的显卡型号以及是否有独立显卡，如果没有以 “NVIDIAN” 名称开头的独立显卡，则不能支持 CUDA，因此 GPU 不可用。

然后，你可以在 [这个页面](#) 查询 GPU 是否支持 CUDA。如果你的 GPU 型号在页面的列表中，则表示你的计算机搭载了能够利用 CUDA 加速应用的现代 GPU，否则 GPU 也不可用。

若 GPU 支持 CUDA，你还需要确保已经完成了上面介绍过的 CUDA 工具包的安装。

2. 检查显卡驱动版本。

在终端中输入 “nvidia-smi” 命令，会显示出显卡驱动的版本和 CUDA 的版本，如下图所示。

```
(pytorch) fmorenovr@ERJ0700:/data/fmorenovr/github/LearnToPayAttention$ nvidia-smi
Mon Dec 14 21:48:47 2020

+-----+
| NVIDIA-SMI 455.45.01      Driver Version: 455.45.01      CUDA Version: 11.1     |
+-----+-----+
| GPU   Name                Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan   Temp   Perf          Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|                           |                    |            MIG M.     |
+-----+-----+
|  0   Quadro RTX 6000       On          | 00000000:15:00.0 Off |          0%      Off |
| 34%   35C    P8           6W / 260W    |  5MiB / 24220MiB |             Default   |
|                           |                    |            N/A       |
+-----+-----+
|  1   Quadro RTX 6000       On          | 00000000:2D:00.0 Off |          0%      Off |
| 33%   38C    P8          16W / 260W    |  72MiB / 24211MiB |             Default   |
|                           |                    |            N/A       |
+-----+-----+

+-----+
| Processes:                                                       GPU Memory |
|  GPU   GI    CI          PID    Type   Process name                  Usage      |
+-----+-----+
|  0      N/A  N/A         1404     G   /usr/lib/xorg/Xorg              4MiB      |
|  0      N/A  N/A         1615     G   /usr/bin/gnome-shell            0MiB      |
|  1      N/A  N/A         1404     G   /usr/lib/xorg/Xorg             64MiB      |
|  1      N/A  N/A         1615     G   /usr/bin/gnome-shell            6MiB      |
+-----+-----+
```

如果显卡驱动的版本过低，与 CUDA 版本不匹配，那么 GPU 也不可用，需要根据显卡的型号更新显卡驱动。

我用表格的方式，帮你梳理了 CUDA 版本与 GPU 驱动版本的对应关系，你可以根据自己计算机驱动的情况对照查看。例如，CUDA 11.1 支持的 Linux 驱动程序是 450.80.02 以上。

CUDA 工具包版本	CUDA所需最低驱动程序版本	
	Linux x86_64 驱动程序版本	Windows x86_64 驱动程序版本
CUDA 11.1	>=450.80.02	>=456.38
CUDA 10.2	>= 440.33	>= 441.22
CUDA 10.1	>= 418.39	>= 418.96
CUDA 10.0	>= 410.48	>= 411.31
CUDA 9.2	>= 396.26	>= 397.44

我们可以在 [🔗这里](#) 下载并安装显卡驱动程序。

3. 检查 PyTorch 版本和 CUDA 版本是否匹配？

PyTorch 版本同样与 CUDA 版本有对应关系，我们可以在 [🔗这个页面](#) 查看它们之间的对应关系。如果两者版本不匹配，可以重新安装对应版本的 PyTorch，或者升级 CUDA 工具包。

使用 Docker

通过 Docker 使用 PyTorch 也非常简单，连安装都不需要，但是前提是你需要熟悉有关 Docker 的知识。

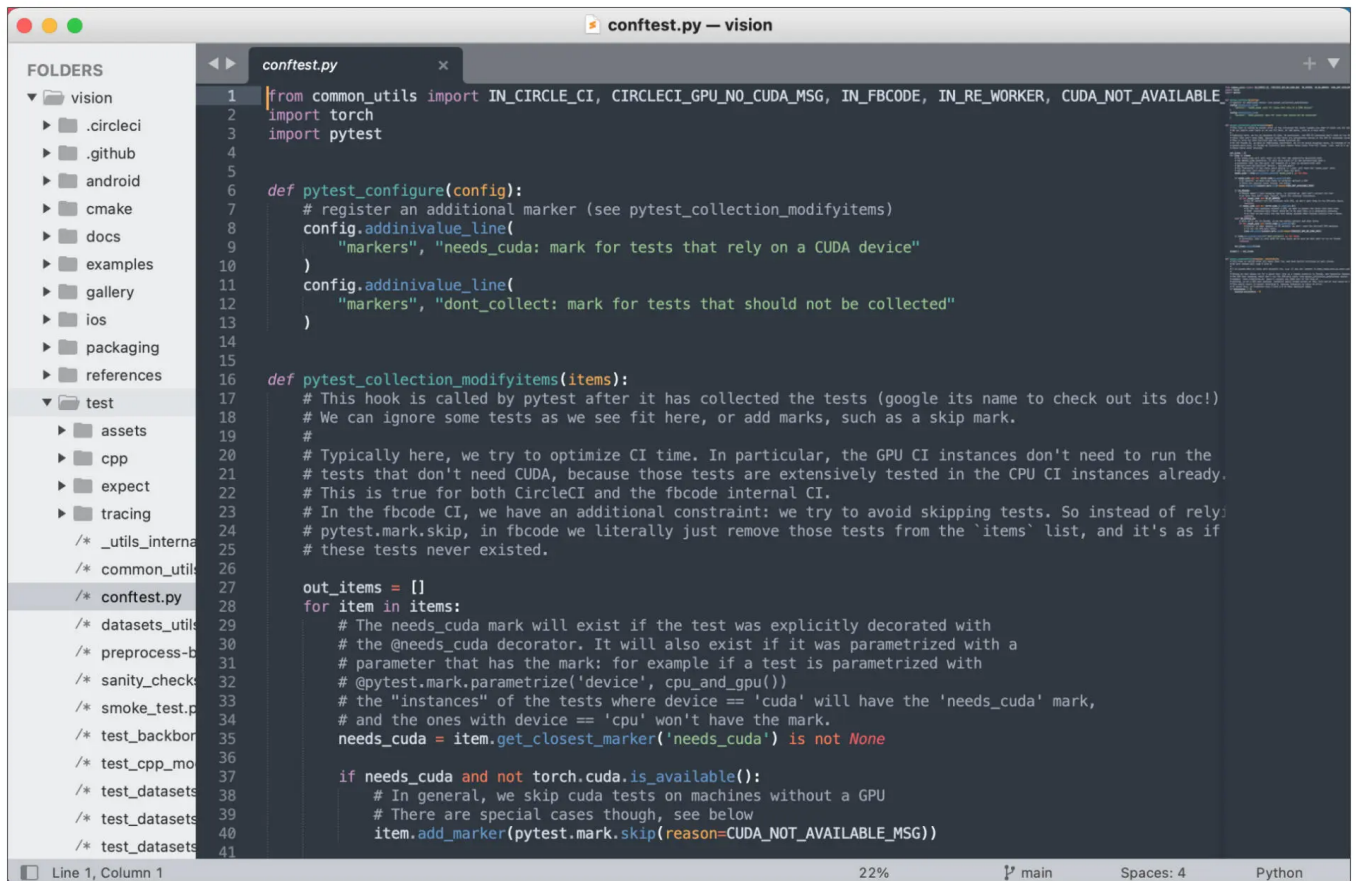
如果你会熟练地使用 Docker，我推荐后面这个网页链接供你参考，[🔗这里](#) 有很多的 PyTorch 的 Docker 镜像，你可以找到自己需要的镜像，然后拉取一个镜像到你的服务器或者本地，直接启动就可以了，无需额外的环境配置。

常用编程工具

在使用 PyTorch 进行编程之前，我们先来看看几个常用的编程工具，但是并不要求你必须使用它们，你可以根据自己的喜好自由选择。

Sublime Text

Sublime Text 是一个非常轻量且强大的文本编辑工具，内置了很多快捷的功能，对于我们开发来说非常便捷。



例如，它可以自动为项目中的类、方法和函数生成索引，让我们可以跟踪代码。具体就是通过它的 goto anything 功能，根据一些关键字，查找到项目中的对应的代码行。另外，它能支持的插件功能也很丰富。

PyCharm

PyCharm 作为一款针对 Python 的编辑器，配置简单、功能强大，使用起来省时省心，对初学者十分友好。它拥有一般 IDE 所具备的功能，比如：语法高亮、项目管理、代码跳转、代码补全、调试、单元测试、版本控制等等。



Vim

Vim 是 Linux 系统中的文本编辑工具，非常方便快捷，并且很强大。我们在项目中经常用到它。

在我们的项目中，经常是需要登录到服务器上进行开发的，服务器一般都是 Linux 系统，不会有 Sublime Text 与 PyCharm，所以，我们用 Vim 打开代码，直接去进行编辑就可以了。

对于没有接触过 Linux，或者一直习惯使用 IDE 来编程开发的同学，初步接触的时候，可能觉得 Vim 不是很方便，但实际上，Vim 包含了丰富的快捷键，对于 Shell 与 Python 的开发来说非常高效。

但是 Vim 的缺点正如刚才所说，你需要去学习它的使用方法，有一点点门槛，但是只要你学会了，我保证你将对它爱不释手。

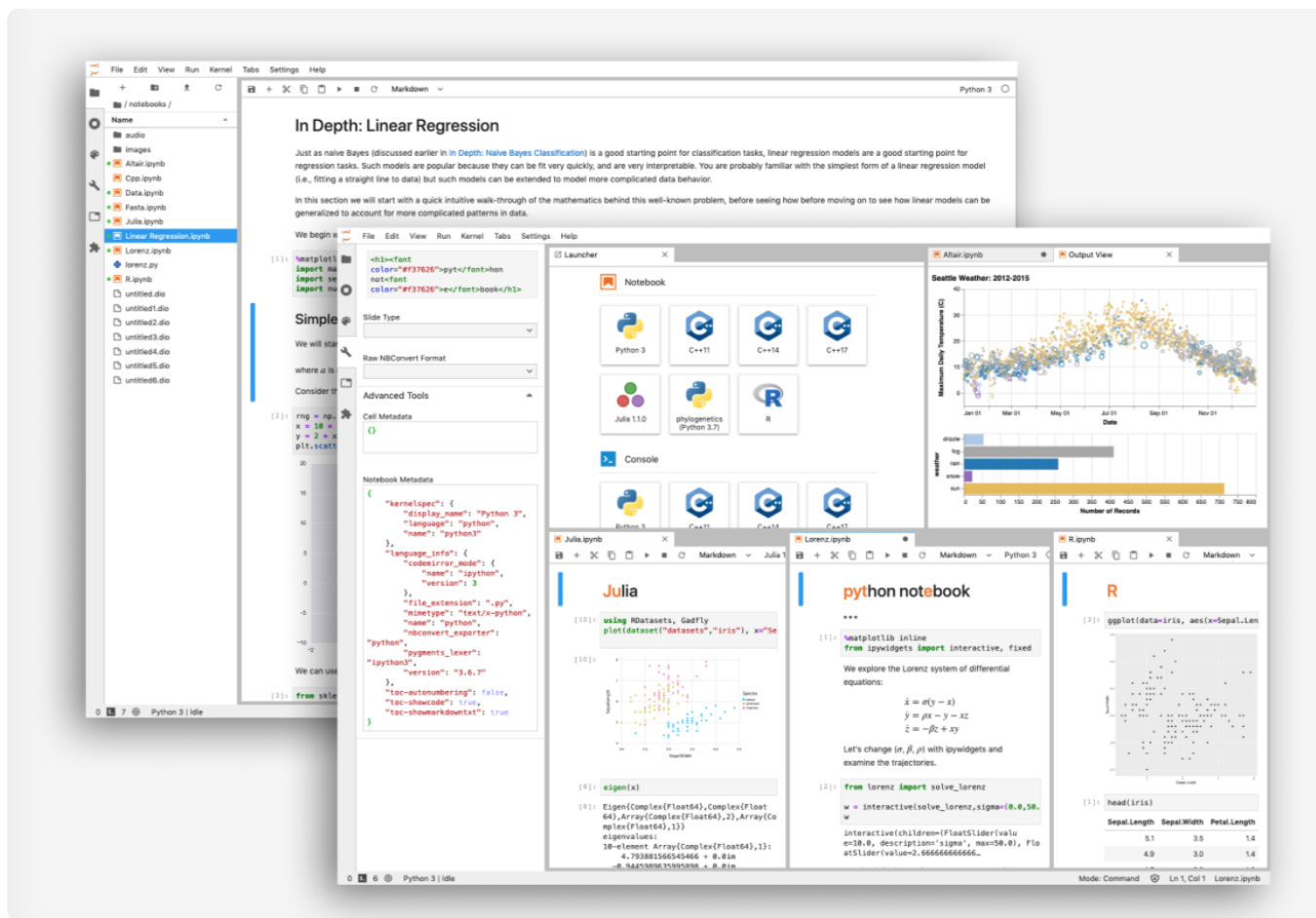
Jupyter Notebook&Lab

Jupyter Notebook 是一个开源的 Web 应用，这也是我最想推荐给你的一个工具。它能够让你创建和分享包含可执行代码、可视化结构和文字说明的文档。

在后面的课程，如果涉及图片生成或结果展示，我们也会使用到 Jupyter Notebook，这里推荐你先安装好。

简而言之，Jupyter Notebook 是以网页的形式打开，可以在网页页面中直接编写代码和运行代码，代码的运行结果也会直接在代码块下显示。比如在编程过程中需要编写说明文档，可以在同一个页面中直接编写，便于及时说明、解释。

而 Jupyter Lab 可以看做是 Jupyter Notebook 的终极进化版，它不但包含了 Jupyter Notebook 所有功能，并且集成了操作终端、打开交互模式、查看 csv 文件及图片等功能。



Jupyter Notebook 在我们的深度学习领域非常活跃。在实验测试阶段，相比用 py 文件来直接编程，还是 Jupyter Notebook 方便一些。在项目结束之后如果要书写项目报告，我觉得用 Jupyter 也比较合适。

使用 pip 安装 Jupyter

通过 pip 安装 Jupyter Notebook 的命令如下。

 复制代码

```
1 pip install jupyter
```

通过 pip 安装 Jupyter Lab 的命令如下。

 复制代码

```
1 pip install jupyterlab
```

启动 Jupyter

完成安装，就可以启动了。我们直接在终端中，执行下面的命令，就可以启动 Jupyter Notebook。

 复制代码

```
1 jupyter notebook
```

启动 Jupyter Lab 需要在终端执行如下命令。

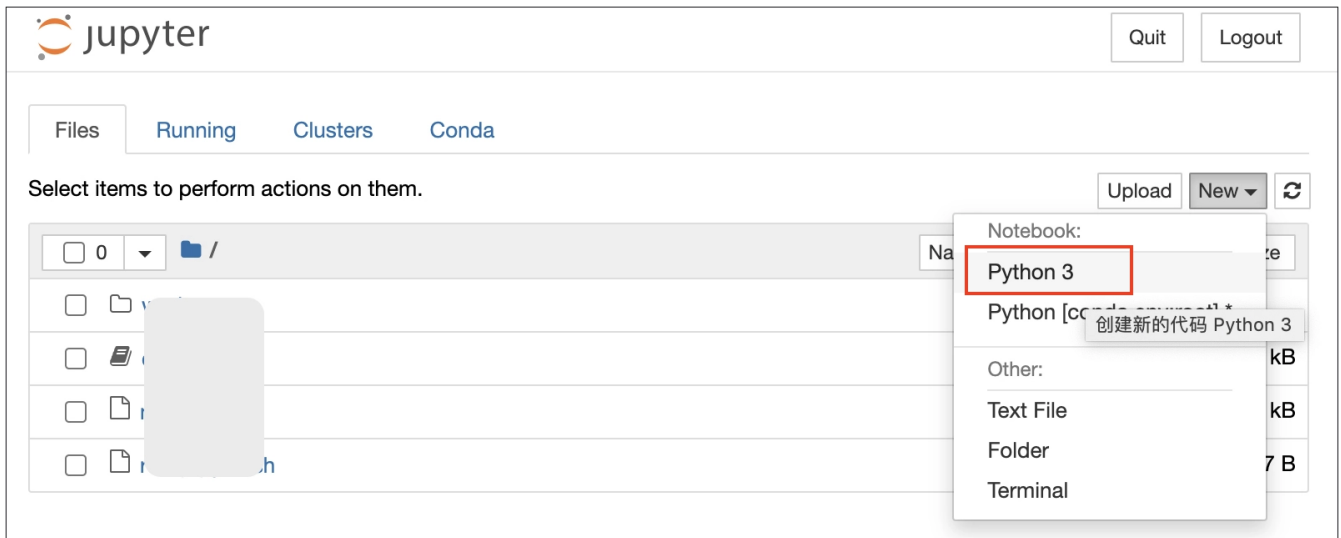
 复制代码

```
1 jupyter lab
```

不管在 macOS 系统里，还是在 Windows 系统，通过以上任意一种方式启动成功之后，浏览器都会自动打开 Jupyter Notebook 或者 Jupyter Lab 的开发环境（你可以回顾下 “Jupyter Notebook & Lab” 那个例子中的界面）。

运行 Jupyter Notebook

进入到 Jupyter Notebook 的界面，我们尝试新建一个 Python 的 Notebook。具体操作方法如下图所示。点击 “New” 下拉菜单，然后点击 “Python 3” 选项，来创建一个新的 Python Notebook。

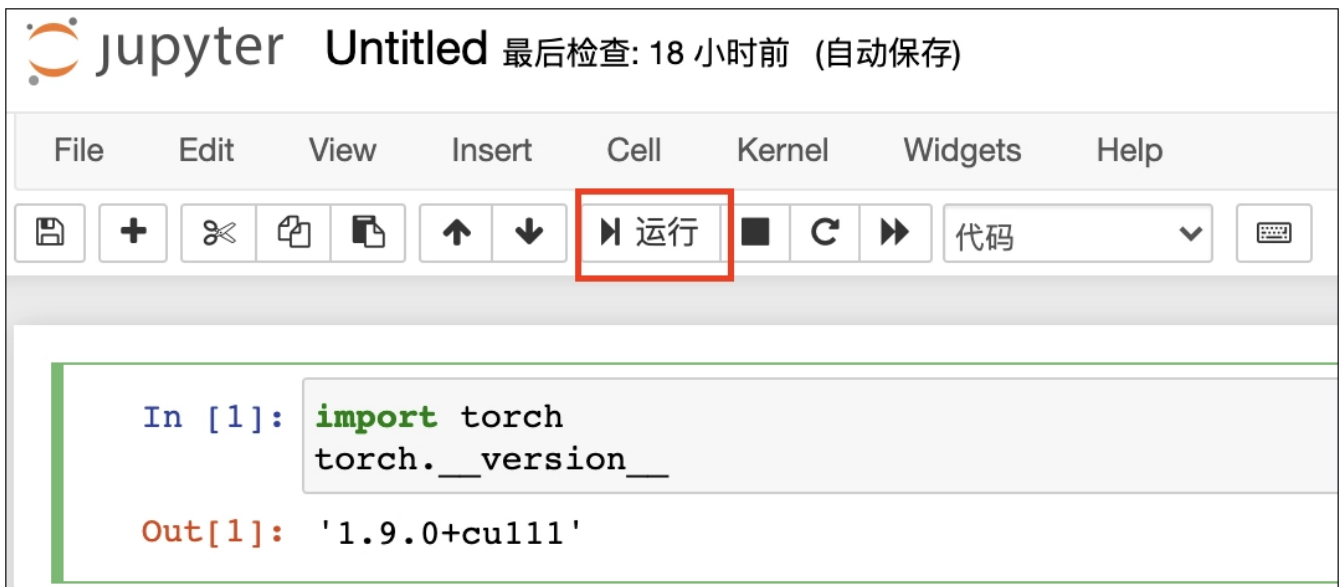


上面我们已经讲过了 PyTorch 的安装方法，我们可以执行下面这段代码，来看看 PyTorch 是否已经安装成功。

复制代码

```
1 import torch
2 torch.__version__
```

点击运行按钮，我们可以看到代码执行的结果，输出了当前安装的 PyTorch 的版本，即 PyTorch 1.9.0 的 GPU 版本。这说明 PyTorch 框架已经安装成功。



小结

恭喜你完成了这节课的学习。

今天，我们一起了解了 PyTorch 框架的用途，简单来说就是能利用 GPU 帮我们搞定深度学习中一系列复杂运算的框架。

想要用好这个工具，我们就得设计好整个任务的流程、整个网络架构，这样 PyTorch 才能实现各种各样复杂的计算功能。

之后我们学习了 PyTorch 框架的安装方法，我还给你推荐了一些深度学习编程的常用工具。**其中我最推荐的工具就是 Jupyter Notebook，这个工具在深度学习领域里常常会用到，后面课程里涉及图片生成或者结果展示的环节，我们也会用到它。**

课程的准备工作就是这些，让我们一起动手，配置好环境，并选择一个你觉得顺手的开发工具，正式开始 PyTorch 的探索之旅吧！

千里之行，始于足下。我在下节课等你，如果你有什么问题，也可以通过留言区和我交流。

我是方远，我们下一讲见！

分享给需要的人，Ta 订阅后你可得 **20 元现金奖励**

 生成海报并分享

 赞 5  提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 [开篇词 | 如何高效入门 PyTorch ?](#)

下一篇 [02 | NumPy \(上\) : 核心数据结构详解](#)

精选留言 (15)

 写留言



悠闲不自得

2021-10-12

请问老师，这个没有视频吗？

展开 ∨

编辑回复: 你好，课程是图文加音频形式的哦。



2



Swaggy

2021-10-11

老师可以讲一点网络训练技巧吗，比如如何应对循环神经网络梯度弥散的意思。

作者回复: 你好，Swaggy，谢谢你的留言。本专栏重点还是PyTorch的使用与一些实战的练习，在实战篇里会穿插一些训练技巧，敬请关注。



3



guang384

2021-10-12

期待更新

展开 ∨

编辑回复: 感谢支持，专栏每周一、三、五更新，敬请期待！



1



徐富献

2021-10-14

老师你好，想问下推荐使用anaconda进行安装的方式吗？

作者回复: 你好，徐富献。感谢你的留言。

可以使用anaconda进行安装。其实用什么开发，都是可以的。取决于个人习惯和开发环境。

例如我是要登录服务器上进行开发。

服务器是Linux系统，没有图形界面。所以，我在编程的时候基本是直接使用vim。

如果需要做一些实验性的尝试，或者想观察一些处理后的图片效果这样的，我会在服务器上启动jupyter，然后在我本地的电脑就可以直接查看和尝试了。



**官**

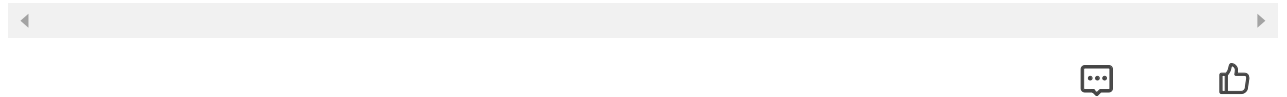
2021-10-14

vscode+jupyter notebook 这个搭配也挺好用的，相比默认的notebook感觉调试方便一些，自己之前经常用sklearn包做一些简单任务的时候受益匪浅

展开 ∨

作者回复: 你好，官。感谢你的留言。👍，等我尝试一下^^。

因为我都是在服务器上开发，一般都使用vim+jupyter的组合。

**西红柿牛男**

2021-10-14

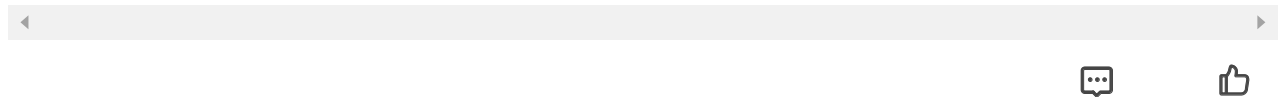
jupyter安装完如何打开？好像没讲，网页地址在哪里找？

作者回复: 你好，西红柿牛男。谢谢你的留言。

在文稿中有说到，在Windows或Mac中，默认情况下执行jupyter notebook或jupyter lab后，都会默认启动浏览器。

执行上述的命令后，也会打印出URL。

你可以看下在执行命令后，打印的“Or copy and paste one of these URLs:”后给出的URL。

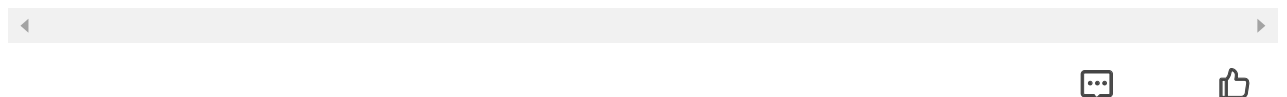
**破绽**

2021-10-13

打卡第一天~从从安装开始~

展开 ∨

作者回复: 你好，破绽。感谢你的留言，一起加油~

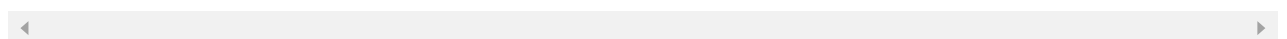
**人间失格**

2021-10-12

希望老师更新快一点，学习可以连贯些。

展开 ∨

编辑回复: 谢谢你的留言，专栏是每周一三五更新的，下一节课明天就出炉啦。



**bbbbbbbbbb**

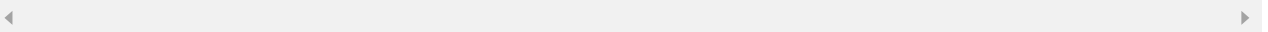
2021-10-12

请问没有显卡，可以有什么不用买显卡的解决方案吗

展开 ∨

作者回复: 你好，谢谢你的留言。是有可以薅羊毛的地方^^。Kaggle每周可以免费使用40个小时的GPU。

我记得还有几个其他的平台可以免费使用GPU，你可以搜一搜“免费GPU”。

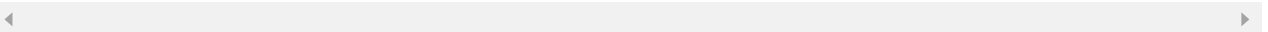
**fengzhiyu_sh**

2021-10-12

能推荐一本市面的pytorch书配套学习么？谢谢

展开 ∨

作者回复: 你好，fengzhiyu_sh。感谢你的留言。我觉得先不着急购买，你可以先看看专栏的内容。我相信如果专栏的内容都搞懂之后，你会有自己解决问题的能力。当遇到了新问题，也会知道从哪个方向去查阅资料来解决问题。

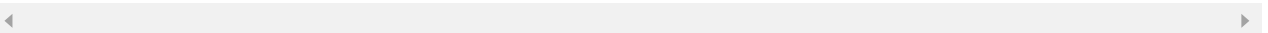
**Abcd**

2021-10-12

AMD显卡可以用吗

展开 ∨

作者回复: 你好，Abcd。感谢你的留言。我还真没尝试过使用AMD的显卡^^，目前主流方式还是使用NVIDIA显卡，CUDA只能与NVIDIA显卡搭配使用。

**夏阳**

2021-10-11

催更催更~

展开 ∨

编辑回复: 每周一三五更新, 敬请期待。



hbuelgr
2021-10-11

直接入手。
展开 ▾

作者回复: 你好, hbuelgr。谢谢你的留言, 一起加油:)。



柱子Prime
2021-10-11

无限期待~
展开 ▾

作者回复: 你好, 柱子, 感谢你的留言, 后面更精彩哦:)。



培基
2021-10-11

好想学, 但零基础, 怕学不会。不敢学!
展开 ▾

作者回复: 你好, 培基。感谢你的留言, 谁都是从0基础过来的, 包括我:), 不需要担心, 没有问题的, 一起加油。